



# Protocolos de Encaminhamento

---



Processo de encaminhamento

---

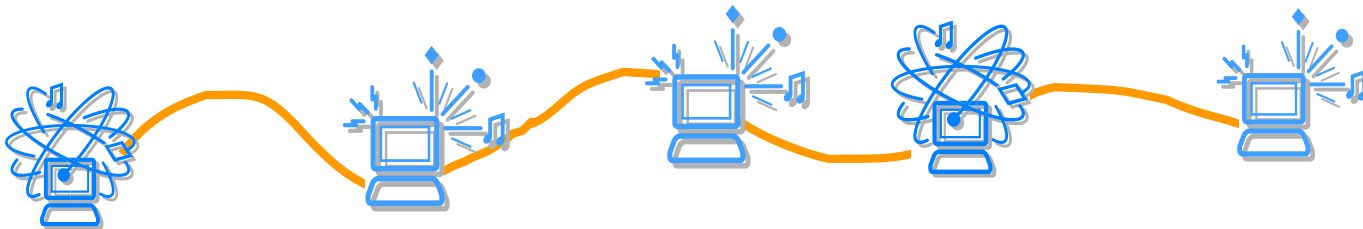


- Endereçamento IP
- Formas de entrega
- Estratégias de encaminhamento
- Algoritmos de selecção de rotas
- Sistemas autónomos



# Endereçamento IP

---





- Os endereços IP têm a dimensão de 32bit (4 bytes)
  - Potencial espaço de endereçamento para  $2^{32}$  máquinas (4.294.967.296)
- Dois tipos base de endereços definidos
  - *Unicast* – Destinatário único
  - *Multicast* (inclui *Broadcast*) – Múltiplos destinatários



- Indicados para aplicações que pretendem a difusão de mensagens
  - Difusão de conteúdos
    - Muito usada para fins multimédia
  - Atingir máquinas que não são conhecidas à partida
    - Processos de descoberta de vizinhos usados nos protocolos
  - Processo escalável de distribuição de informação para muitos receptores
    - A difusão de um canal de TV com qualidade “PAL” em *unicast* necessita na origem de cerca de 6Mbit/s por destinatário
    - Com o uso de *multicast* necessitará apenas dos mesmos 6Mbit/s mas uma única vez, independentemente do número de destinatários

# Endereços *Unicast*



- Para troca de mensagens entre duas máquinas
- Recentemente foi introduzido o conceito de *anycast* proveniente do desenvolvimento do IPv6
  - Endereços pertencentes aos blocos de endereçamento *unicast*
  - Não podem servir para iniciar ligações
  - Usado em situações em que se pretende atingir a máquina mais próxima que presta o serviço
  - Actualmente em uso na Internet
    - DNS – Root Servers (ex. f.root-servers.net )
  - Na IPLNet
    - NTP – Actualização de relógios ( ntp.net.ipl.pt )
    - DNS – Servidores de resolução de endereços ( forwarders )

# Hierarquização a dois níveis

---



- Só os endereços *unicast* têm estas duas componentes
- À semelhança de rede telefónica
  - A componente de maior peso identifica a rede onde a máquina está
    - Os encaminhadores do tráfego (*routers*) só precisam dela para tomar decisões ( analogia: em qualquer país +351... é para Portugal )
  - A componente de menor peso identifica a máquina dentro da rede
  - Uma diferença de salientar em relação à rede telefónica
    - Não existe uma relação rígida entre os identificadores de rede e a sua localização geográfica
  - Para acomodar diferentes necessidades de dimensão de rede o espaço de endereçamento *unicast* foi subdividido em classes com diferentes dimensões da componente **rede e máquina**

# Classes de endereços



	8	16	24	32
Classe A	0	Rede	Máquina	
Classe B	10	Rede	Máquina	
Classe C	110	Rede	Máquina	
Classe D	1110	Multicast		
Classe E	1111	Experimental		

- Classe A – *unicast*
  - $2^7$  (127) redes com  $2^{24}$  (16M) máquinas cada
- Classe B – *unicast*
  - $2^{14}$  (16K) redes com  $2^{16}$  (64K) máquinas cada
- Classe C – *unicast*
  - $2^{21}$  (2M) redes com  $2^8$  (256) máquinas cada
- Classe D – *multicast*
  - $2^{28}$  grupos (endereços de multicast)
- Classe E – *experimental*
  - $2^{28}$  endereços sem comportamento definido





# Notação e manipulação

---

- Os 32 bit do endereço são representados em decimal, byte-a-byte, do maior para o menor peso, separados pelo caracter ponto (.)
  - Exemplo: 192.0.2.100 = 11000000 00000000 00000010 01100100
- As operações sobre endereços são normalmente operações lógicas realizadas bit-a-bit

# Classes na notação

---



Classe	Início	Fim
A	1.0.0.0	126.255.255.255
B	128.0.0.0	191.255.255.255
C	192.0.0.0	223.255.255.255
D	224.0.0.0	239.255.255.255
E	240.0.0.0	255.255.255.254

# Endereços reservados



- Diversas porções do espaço de endereçamento estão reservados para fins específicos (revisão mais recente no RFC3330)

Address Block	Present Use	Reference
0.0.0.0/8	"This" Network	[RFC1700, page 4]
10.0.0.0/8	Private-Use Networks	[RFC1918]
14.0.0.0/8	Public-Data Networks	[RFC1700, page 181]
24.0.0.0/8	Cable Television Networks	--
39.0.0.0/8	Reserved but subject to allocation	[RFC1797]
127.0.0.0/8	Loopback	[RFC1700, page 5]
128.0.0.0/16	Reserved but subject to allocation	--
169.254.0.0/16	Link Local	--
172.16.0.0/12	Private-Use Networks	[RFC1918]
191.255.0.0/16	Reserved but subject to allocation	--
192.0.0.0/24	Reserved but subject to allocation	--
192.0.2.0/24	Test-Net	
192.88.99.0/24	6to4 Relay Anycast	[RFC3068]
192.168.0.0/16	Private-Use Networks	[RFC1918]
198.18.0.0/15	Network Interconnect Device Benchmark Testing	[RFC2544]
223.255.255.0/24	Reserved but subject to allocation	--
224.0.0.0/4	Multicast	[RFC3171]
240.0.0.0/4	Reserved for Future Use	[RFC1700, page 4]

# Endereçamento IP - Endereços especiais



Rede	Máquina	Uso
<Valor>	tudo "0"	Endereço de rede
<Valor>	tudo "1"	Endereço de broadcast directo a uma rede [1]
tudo "1"		Broadcast limitado à rede local
tudo "0"		A própria máquina [2]
tudo "0"	<Valor>	Uma máquina na mesma rede [2]
127	<Qualquer>	Loopback
[1] É recomendável os <i>routers</i> não encaminharem este tráfego		
[2] Situações de uso abandonado, para compatibilidade		

- Limita na prática o número de endereços disponíveis para máquinas a  $2^n - 2$  por rede (com “n” – sendo o número de bits da componente máquina atribuídos à rede em questão)

# Problemas do endereçamento IP clássico (1)

---



- Quando foi concebido o espaço de endereçamento IP (32 bits) pensava-se que era impossível de esgotar
  - Face ao número de máquinas e redes que existiam na altura
- Rapidamente se percebeu que não era bem assim ...
  - O encaminhamento IP obriga a que cada rede física tenha um endereço de rede diferente
  - Com a popularização da Internet o número de máquinas e redes cresceu exponencialmente
  - O esquema de classes de endereços é muito pouco flexível
    - Provoca um grande desperdício de endereços

# Problemas do endereçamento IP clássico (2)

---

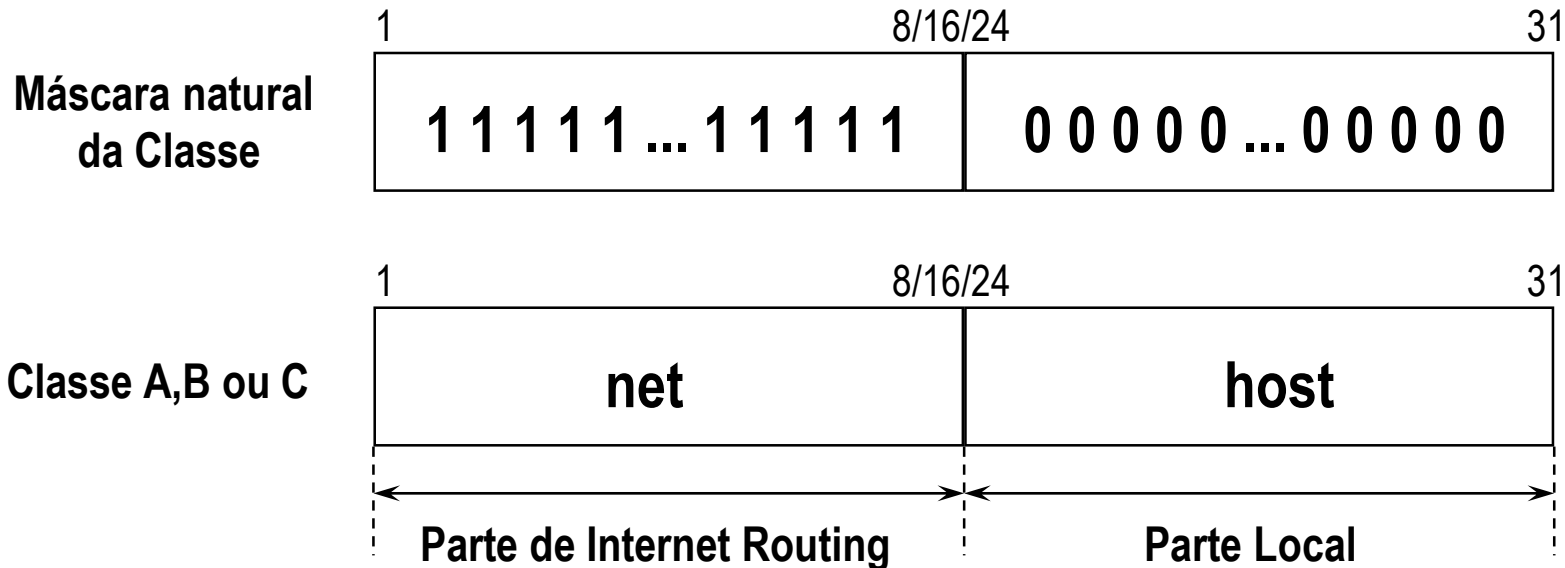


- O esquema de classes de endereços provoca um grande desperdício de endereços
  - Uma rede com mais de 254 máquinas necessita de usar um endereço de rede de classe B
    - Endereços de rede classe B são os mais requisitados
  - Há muitas redes com mais de 254 máquinas mas com muito menos que 65.534 endereços de máquinas
    - Endereços de rede classe B subaproveitados
  - Não há (muitas) redes com 16.777.214 de máquinas
    - Endereços de rede classe A subaproveitados



# Máscara de rede (*Subnet Mask*)

- Define onde se situa a divisão do endereço IP em parte de rede (*net*) e parte de máquina (*host*)
  - A máscara tem “1” nos bits que correspondem à parte do endereço que identifica a rede e “0” nos bits que correspondem à parte do endereço que identifica a máquina



# Representação da Máscara de rede

---



- Notação de pontos (usada nos endereços)
  - Ex.: 192.0.2.0 255.255.255.0
    - Indicação dos bits da máscara em numeração decimal
- Notação CIDR (*Classless Inter-Domain Routing*)
  - Ex.: 192.0.2.0 / 24
    - Indicação do números de bits a 1 existentes na máscara



# Subnet Addressing (Sub-Redes IP)

---



- Mecanismo que permite distribuir uma classe de endereços IP por várias redes físicas (sub-redes IP)
  - Utiliza o conceito de máscara de rede
  - Tem que haver cooperação entre máquinas e *routers*
  - Só são visíveis pelas máquinas e *routers* da rede
  - Não são visíveis pelos routers de fora da rede
  - Definido no RFC 950 [1985]

# Conceito de sub-rede (*Subnet*)



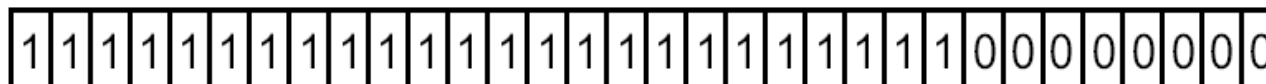
- A classe do endereço IP identifica implicitamente os bits da componente de rede e a máscara divide a parte de sub-rede (*subnet*) e parte de máquina (*host*)
  - Ex.: subdivisão de uma classe B em várias redes de dimensão equivalente a classes C

IP address



Apply the subnet mask

Subnet mask



Result = subnet/host boundary



# Operações usando a máscara de rede



- Para obter o endereço de rede ou de *broadcast* dirigido a partir de um endereço qualquer da rede e a máscara desta:

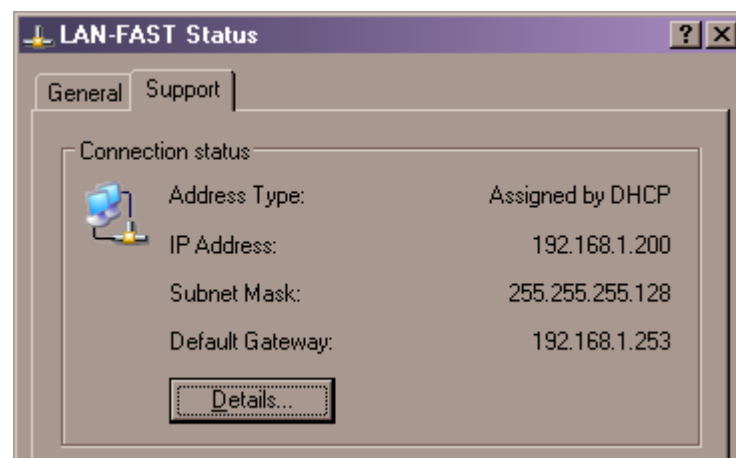
Rede = IP AND Mask

AND	192 . 168 . 1 . 200	200= 11001000
	255 . 255 . 255 . 128	128= 10000000
<hr/>		
Rede= 192 . 168 . 1 . 128		

Broadcast = IP AND NOT Mask

OR	192 . 168 . 1 . 200	200= 11001000
	0 . 0 . 0 . 127	127= 01111111
<hr/>		
Broadcast= 192 . 168 . 1 . 255		

Operações mais comuns	
X AND 0 = 0	X OR 255 = 255
X AND 255 = X	NOT 0 = 255
X OR 0 = X	NOT 255 = 0



# Variable Length Subnet Mask (VLSM)

---



- **Subnetting:** especifica uma máscara que divide o bloco de endereços definido pela máscara natural (i.e. classe A, B C) em subredes iguais
  - Uma única máscara de rede (ex.: 255.255.255.0) é usada para todas as subredes
- **VLSM:** Múltiplas máscaras definem subredes de diferentes dimensões no espaço de endereçamento de uma classe.
  - Ex: 255.255.255.128 e 255.255.255.224 podem ser usadas para dividir o espaço de endereçamento de uma rede classe C
  - Permite o uso mais eficiente do espaço de endereçamento.

# Prefixos de 31 Bit em ligações IPv4

---



- Objectivo
  - Uso blocos de 2 endereços (/31) em ligações ponto-a-ponto - em vez de blocos de 4 endereços (/30) - para poupar
- Características
  - Os 2 endereços são considerados de “host”
    - Atribuídos às máquinas nas extremidades da ligação
  - O endereço de sub-rede é o mais baixo
  - Não existe endereço de “directed broadcast” para a sub-rede
    - Ligações ponto-a-ponto não há *broadcast*
  - Ainda não suportado genericamente pelos equipamentos
  - Aumenta a eficiência da utilização do espaço de endereçamento de 50% para 100%
  - Documentado no RFC 3021 [12/2000]

# Superneting (1)



- Mecanismo que permite agregar vários blocos de endereçamento para usar numa única rede (RFC1338)
  - Os blocos a agregar têm de ser contíguos
  - A dimensão agregada tem de ser uma potência de 2
  - A rede resultante tem de se iniciar numa posição do espaço de endereçamento múltipla da dimensão do bloco
- Pretende resolver problema da falta de endereços IP
  - Antes, redes com mais que 254 máquinas tinham que usar um endereço de rede de classe B (desperdiçando muitos endereços)
  - Com Superneting, redes com até 510 máquinas podem usar dois blocos de endereçamento de classe C como uma só rede IP
  - As tabelas de encaminhamento dos routers exteriores só precisam de ter uma entrada para cada super-rede
    - As redes englobadas não necessitam de ser anunciadas individualmente à Internet

# Superneting (2)

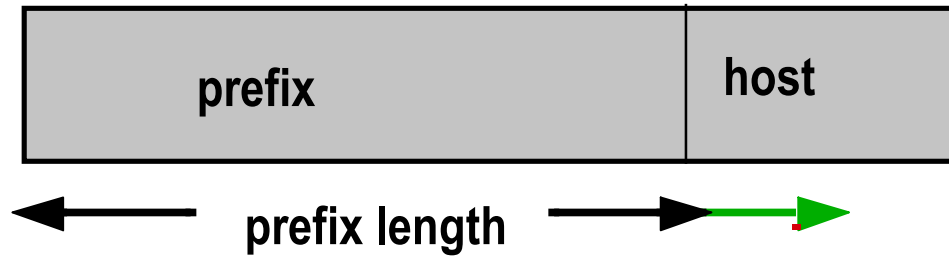


- Os *Internet service providers* (ISP) atribuem a uma entidade um número N de endereços contíguos e com o mesmo prefixo (bits de maior peso iguais para ser possível aglomerar/sumariar)
- Ex.: Supondo que foi atribuído um bloco de 1024 endereços a partir do endereço 193.137.128.0

Decimal	Equivalente binário
193.137.128.0	11000001 10001001 100000 <u>00</u> <u>00000000</u>
193.137.131.255	11000001 10001001 100000 <u>11</u> <u>11111111</u>

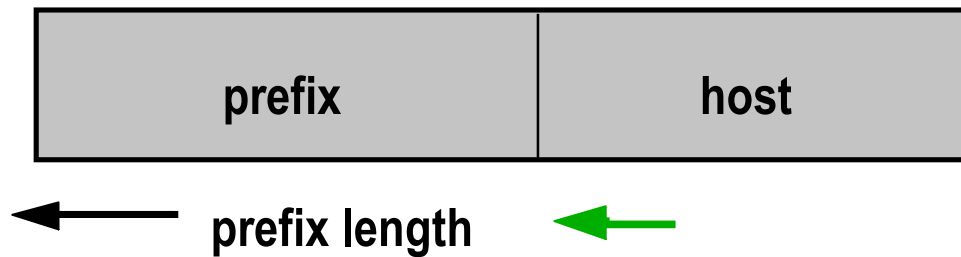
**193.137.128.0 255.255.252.0 - Com máscaras**  
**ou**  
**193.137.128.0 / 22 - CIDR**

# Sumarização de Rotas (*Route Aggregation*)



## Subnetting:

Criação de múltiplas sub-redes tornando o prefixo maior



## Sumarização:

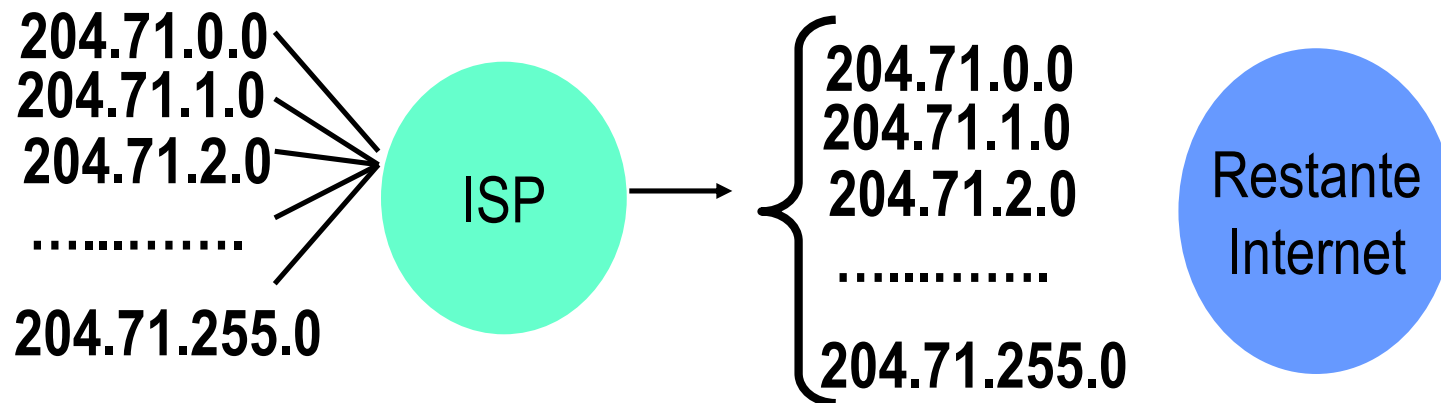
Agregação de múltiplas sub-redes tornando o prefixo mais pequeno (para simplificar as tabelas de encaminhamento)



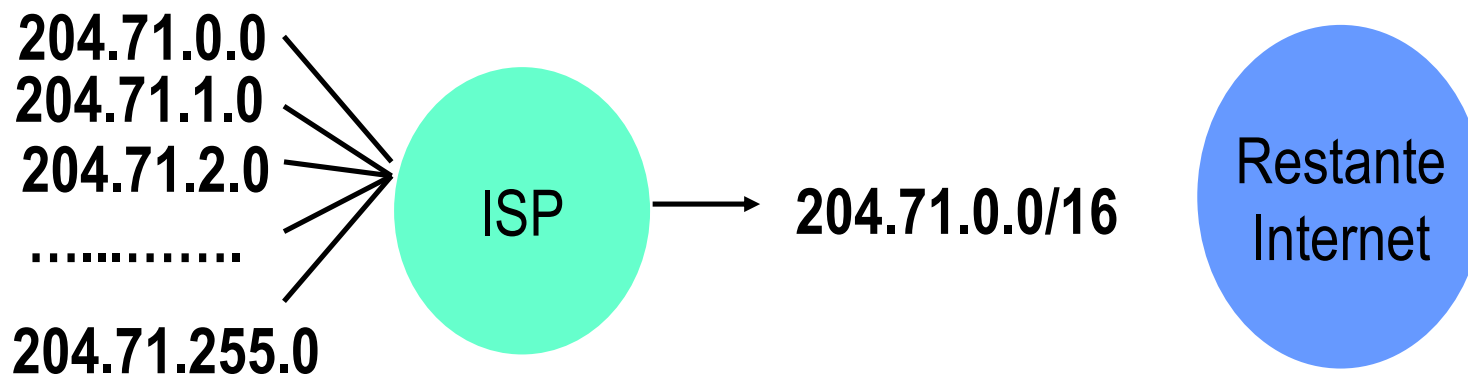
# Sumarização e CIDR



- Sem Sumarização



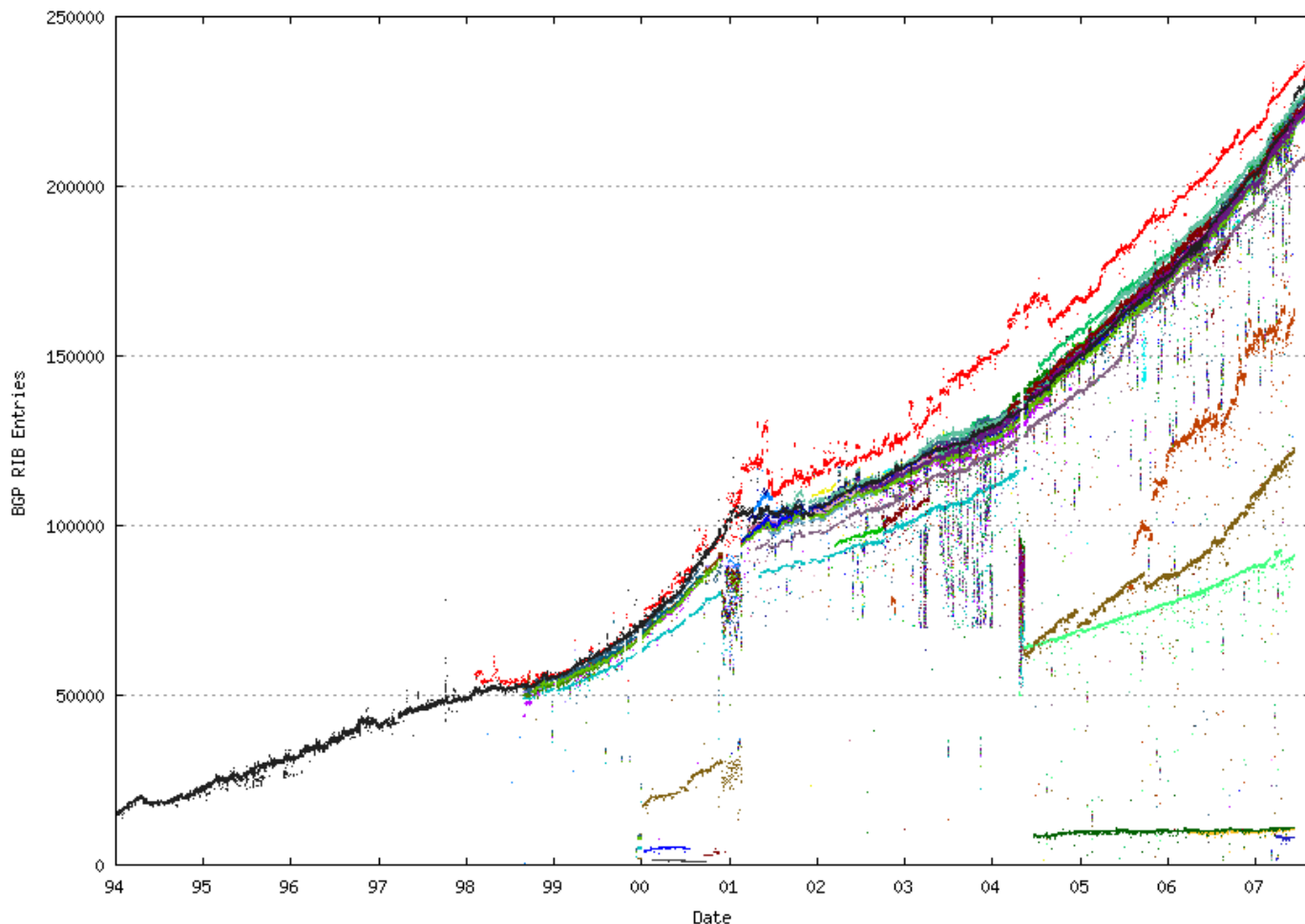
- Com Sumarização



# A importância da sumarização



- Evolução das tabelas de encaminhamento centrais da Internet desde 1994



# CIDR – Classless Inter-Domain Routing



- Definição:  
“*an Address Assignment and Aggregation Strategy*”
- Funcionamento
  - Permite englobar endereços IP em gamas (prefixos)
  - Propõe uma estratégia de atribuição de endereços de rede tendencialmente de acordo com a topologia física e geográfica
    - Atribuição por continente, país, ISP, etc
- Vantagens
  - Permite diminuir as tabelas de encaminhamento dos *routers* que só precisam de ter uma entrada para cada gama
  - Definido nos RFC 1518 e 1519 [Set 1993]  
(torna obsoleto o RFC 1338 - *Supernetting*)

# Agregação de endereços com CIDR

---



- Para se poder aproveitar a capacidade de agregação de endereços é necessário que estes sejam atribuídos de forma organizada.
- Ex.:  
Europa 194.0.0.0 a 195.255.255.255 - 194.0.0.0/7  
EUA 198.0.0.0 a 199.255.255.255 - 198.0.0.0/7
- Simplifica as tabelas de encaminhamento:
  - Nos routers na Europa basta uma entrada na tabela para encaminhar para os EUA e vice-versa. (utopia ...)

# CIDR – *Classless Inter-Domain Routing*

---

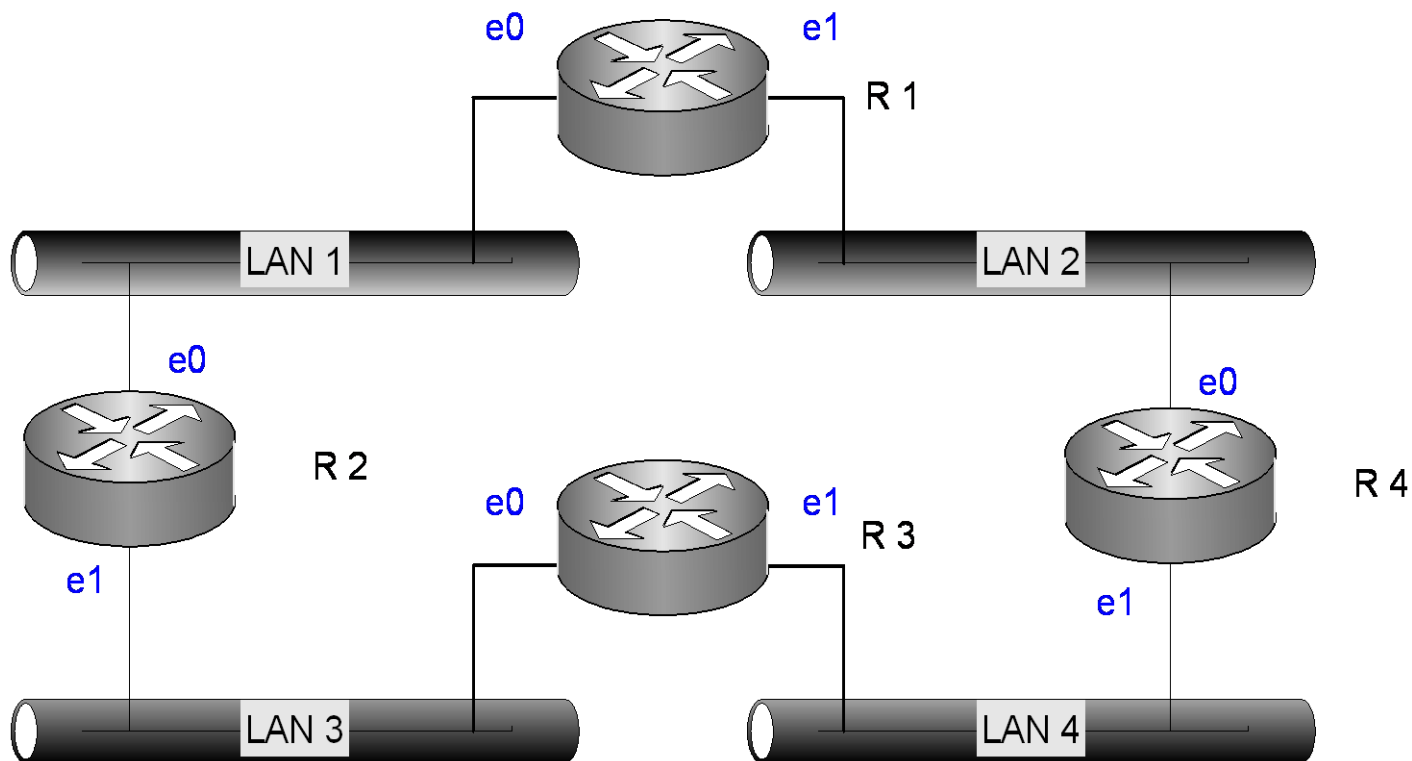


- Problemas
  - Organizações que são *multi-homed* (ligadas a vários ISP)
  - Organizações que mudam de ISP mas não de endereçamento

# Exercício



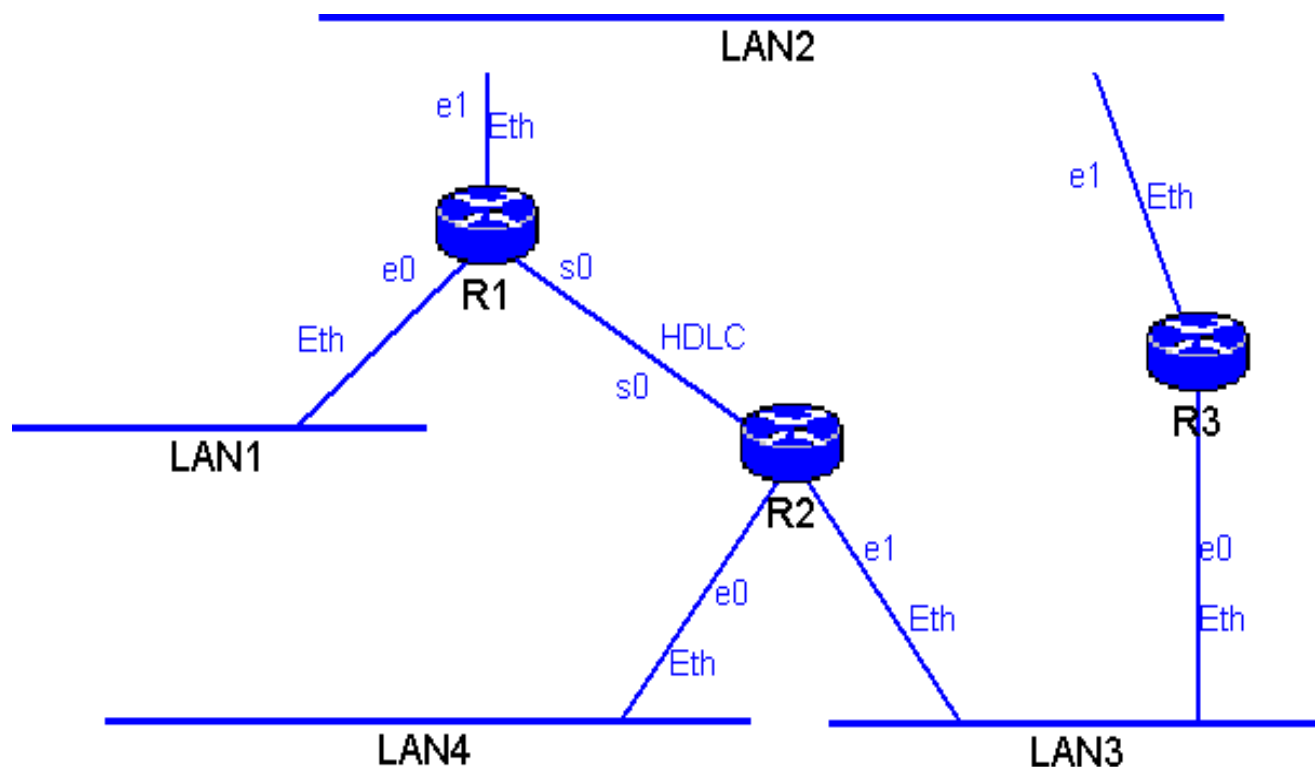
- Distribuição de endereço classe C por 4 sub-redes



# Exercício: (VLSM)



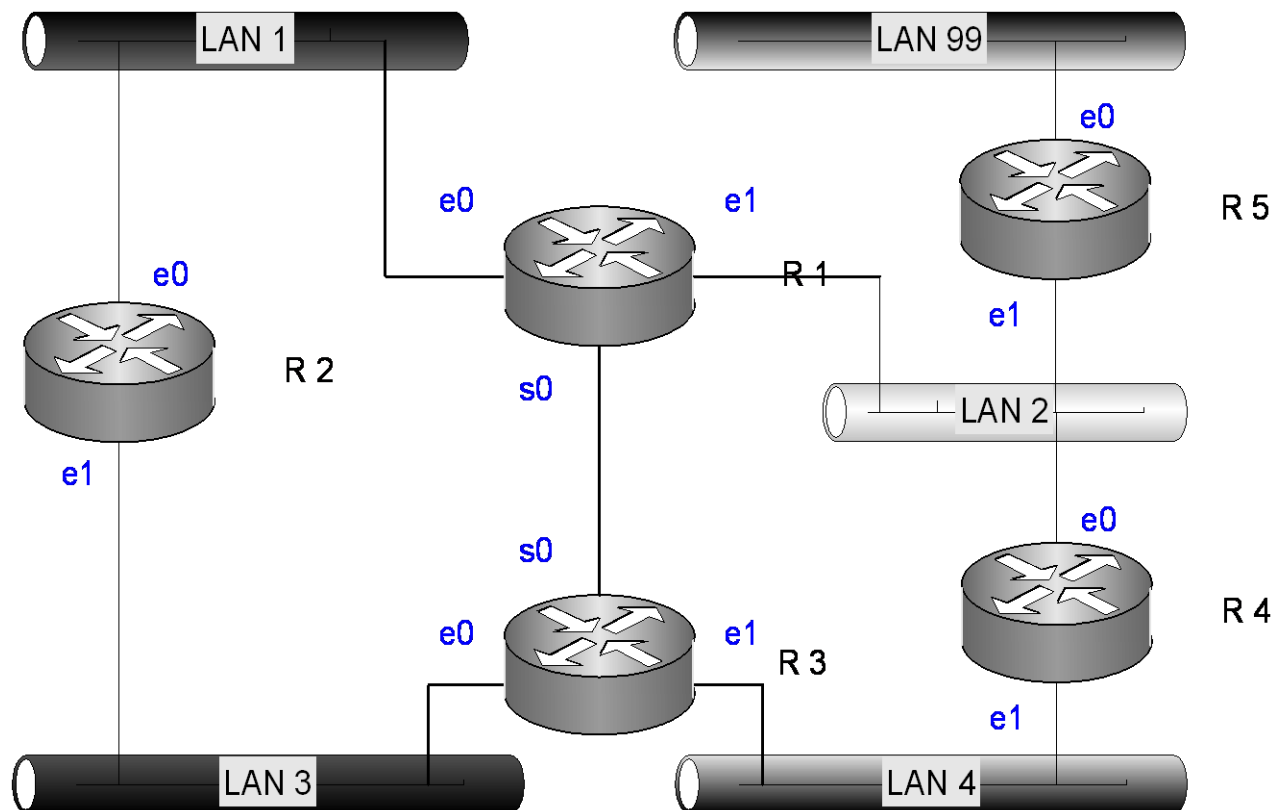
- Distribuição de um endereço classe C pelas LAN otimizando o número de endereços



# Exercício: (VLSM)



- Distribuição do bloco 10.2.3.0/25 pelas LAN otimizando o número de endereços







# Entrega Direta e Indireta de Datagramas

## Tabelas de Encaminhamento

---





- Características
  - Baseado em tabelas de encaminhamento (*routing*)
  - Efectuado pelos *routers* e pelas máquinas
  - Complexidade das tabelas preferencialmente nos *routers*
    - Máquinas - Caminho por omissão (ou o destino é adjacente ou envia para um router – *default gateway*)
  - Configuração de encaminhamento estática (manual) ou dinâmica (automática - protocolos)
  - Máquinas com várias interfaces de rede podem fazer encaminhamento (*routers*)

# Que tramas aceita uma placa de rede *ethernet*?

---



- Em modo promíscuo – Todas
  - Usado por analisadores de protocolos e *bridges/switches*
- Em modo normal de funcionamento
  - Destinadas a si (com endereço MAC destino igual ao seu)
  - Com endereço destino de *broadcast* (48 bit a “1” => ff:ff:ff:ff:ff:ff)
  - Com endereço destino igual a um dos endereços de grupo (*multicast*) a que a placa está receptiva
    - As placas têm normalmente uma tabela que permite que as aplicações que usam *multicast* lá coloquem o endereço de grupo que lhes interessa
    - Por limitações de *hardware* algumas placas de rede não têm esta tabela e para receberem *multicast* têm de ficar receptivas a todas as tramas destinadas a endereços de grupo

# Uso de *routers* como intermediários

---



- Como conseguir que o *router* intermediário aceite um datagrama IP que não é destinado a si?
- O modelo de camadas foi inventado para isso!
  - *Datalink* – Trata da entrega entre nós adjacentes
  - IP – Trata do encaminhamento ao longo da rede
- Ao nível *datalink* (MAC) a mensagem é destinada ao *router* mas ao recebe-la este, como verifica que o endereço IP destino não é local, procura na sua tabela de encaminhamento e reenvia-a para o próximo nó

# Entrega Directa

---



- Quando a tabela de encaminhamento indica que se pode atingir imediatamente o endereço IP destino, enviando para um nó adjacente
- É usado o protocolo ARP (explicado à frente) para determinar o endereço MAC associado ao endereço IP destino do datagrama
- O datagrama é enviado encapsulado numa trama com o endereço MAC destino (da máquina destino) determinado pelo passo anterior

# Entrega Indirecta

---



- Quando a tabela de encaminhamento indica que para atingir o endereço IP destino do datagrama tem de se usar um *router* como intermediário
- É usado o protocolo ARP (explicado à frente) para determinar o endereço MAC associado ao endereço IP indicado na tabela como o intermediário
- O datagrama é enviado sem qualquer alteração dos endereços IP encapsulado numa trama com o endereço MAC destino (do *router*) determinado pelo passo anterior



- Entrega directa
  - Quando a máquina destino está na mesma rede física (IP) da máquina origem
  - No envio do datagrama tanto o endereço IP destino como o endereço MAC destino (da trama que o transporta) pertencem à máquina alvo
  - Na comunicação entre duas máquinas existe sempre uma entrega directa
- Entrega indirecta
  - Quando a máquina destino não está na rede física (IP) da máquina origem
  - No envio do datagrama este é encapsulado numa trama com o endereço MAC destino do *router* escolhido pela tabela de encaminhamento
  - O conteúdo do datagrama não é alterado em trânsito! (excepto TTL e *checksum*)
  - Na comunicação entre duas máquinas poderão existir 0 ou mais entregas indirectas
  - Ao longo do caminho irão existir várias tramas distintas, uma em cada troço de rede.

# Cuidado com as ambiguidades nestas questões...



- Temos tendência para associarmos uma relação não ambígua entre máquina - IP, router – IP, router – MAC, etc
  - «envia para o endereço IP da máquina» - Ambíguo
  - «qual é o endereço MAC do *router*?» - Ambíguo
- Qualquer máquina/*router* pode ter:
  - Múltiplas interfaces/placas de rede
  - Cada interface *Ethernet* tem um endereço MAC gravado de fábrica (possível de alterar, embora não seja usual fazê-lo)
  - Cada interface pode ter um ou mais endereços IP
- Sem ambiguidade
  - «envia para o endereço IP da interface eth2 da máquina»
  - «qual é o endereço MAC da interface *Ethernet0* do *router*?»





- Parâmetros base essenciais
  - Rede destino
    - Definida pelo endereço de rede e máscara
  - Próximo *router*
    - *Router* a ser usado como intermediário na entrega, caso seja directa, neste campo estará 0.0.0.0 ou o endereço IP da interface local de saída
  - Interface
    - Interface local a ser usada no encaminhamento definida com o seu nome ou endereço IP
- Parâmetros adicionais (dependem da implementação)
  - Métrica
  - *Flags* de estado (ex. origem da informação estática/dinâmica)
  - MTU do percurso

# Processo de decisão de encaminhamento (1)

---



- As tabelas de encaminhamento são listas ordenadas por ordem decrescente do número bits activos na máscara
  - Ex. uma rota com máscara 255.255.255.0 (/24) é avaliada antes de outra com máscara 255.255.0.0 (/16)
  - As rotas mais específicas são avaliadas primeiro pois têm máscaras com mais bits activos
- O *router* usa o endereço destino do datagrama a encaminhar como chave da pesquisa na lista
  - Se chegar ao fim da lista (não encontra rota) descarta o datagrama e gera uma mensagem ICMP *network-unreachable* destinada ao IP origem do datagrama original e incluindo uma amostra do datagrama (cabeçalho IP+8bytes) no corpo desta

# Processo de decisão de encaminhamento (2)



- Para cada entrada da tabela (rota) o *router* faz a operação:
  - Se **IPdestino** (do datagrama a encaminhar) AND **Máscara** == **Rede**, encaminha-o pelo router indicado na coluna de próximo router ou faz a entrega directamente se o valor deste for 0.0.0.0 ou igual ao endereço da interface de saída
  - Caso contrário repete o processo para a rota seguinte

Kernel IP routing table

Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
193.137.220.251	193.137.237.173	255.255.255.255	UGH	0	0	0	eth2
193.137.220.252	193.137.237.173	255.255.255.255	UGH	0	0	0	eth2
193.137.237.160	0.0.0.0	255.255.255.240	U	0	0	0	eth2
193.137.220.0	0.0.0.0	255.255.255.128	U	0	0	0	bond0
192.68.221.0	193.137.220.126	255.255.255.0	UG	0	0	0	bond0
192.104.48.0	193.137.220.126	255.255.255.0	UG	0	0	0	bond0
193.137.237.0	193.137.220.126	255.255.255.0	UG	0	0	0	bond0
193.137.220.0	193.137.220.126	255.255.254.0	UG	0	0	0	bond0
10.0.0.0	193.137.220.126	255.0.0.0	UG	0	0	0	bond0
127.0.0.0	0.0.0.0	255.0.0.0	U	0	0	0	lo
0.0.0.0	193.137.237.173	0.0.0.0	UG	0	0	0	eth2

# Exercício: Tabelas de encaminhamento (1)



- Interprete cada uma das entradas da tabela de encaminhamento apresentada retirada de uma máquina *Windows XP*

NetworkDestination	Netmask	Gateway	Interface	Metric
0.0.0.0	0.0.0.0	10.100.255.254	10.100.255.99	8
10.100.255.0	255.255.255.0	10.100.255.99	10.100.255.99	10
10.100.255.99	255.255.255.255	127.0.0.1	127.0.0.1	10
10.255.255.255	255.255.255.255	10.100.255.99	10.100.255.99	10
127.0.0.0	255.0.0.0	127.0.0.1	127.0.0.1	1
224.0.0.0	240.0.0.0	10.100.255.99	10.100.255.99	10
255.255.255.255	255.255.255.255	10.100.255.99	10.100.255.99	1

# Exercício: Tabelas de encaminhamento (2)



NetworkDestination	Netmask	Gateway	Interface	Metric
0.0.0.0	0.0.0.0	10.100.255.254	10.100.255.99	8
10.100.255.0	255.255.255.0	10.100.255.99	10.100.255.99	10
10.100.255.99	255.255.255.255	127.0.0.1	127.0.0.1	10
10.255.255.255	255.255.255.255	10.100.255.99	10.100.255.99	10
127.0.0.0	255.0.0.0	127.0.0.1	127.0.0.1	1
224.0.0.0	240.0.0.0	10.100.255.99	10.100.255.99	10
255.255.255.255	255.255.255.255	10.100.255.99	10.100.255.99	1

- 1 – Rota *default* via 10.100.255.254
- 2 – Para a rede local fazer entrega directa
- 3 – Para o próprio IP enviar pela interface *loopback*
- 4 – Para o endereço de *broadcast* da classe fazer entrega directa
- 5 – Para a rede *loopback* enviar via interface de *loopback*
- 6 – Para a classe D (multicast) fazer entrega directa
- 7 – Para o endereço *broadcast* local fazer entrega directa

# Endereços IP privados

---



- Endereços reservados para uso em redes privadas
  - Podem ser usados internamente em várias redes privadas
  - Não devem circular na Internet
- 10.0.0.0 - 10.255.255.255 ( 10/8 )
- 172.16.0.0 - 172.31.255.255 ( 172.16/12 )
- 192.168.0.0 - 192.168.255.255 ( 192.168/16 )
- Definidos pela (IANA) *Internet Assigned Numbers Authority* no RFC 1918 [1996]

# Exemplo de Tabela de *routing* (Win2K)



```
C:\WIN2K>route print
```

```
=====
```

## Active Routes:

Network Destination	Netmask	Gateway	Interface	Metric
0.0.0.0	0.0.0.0	141.29.155.254	141.29.155.108	1
127.0.0.0	255.0.0.0	127.0.0.1	127.0.0.1	1
141.29.155.0	255.255.255.0	141.29.155.108	141.29.155.108	1
141.29.155.108	255.255.255.255	127.0.0.1	127.0.0.1	1
141.29.255.255	255.255.255.255	141.29.155.108	141.29.155.108	1
200.0.0.0	255.0.0.0	141.29.155.245	141.29.155.108	3
224.0.0.0	224.0.0.0	141.29.155.108	141.29.155.108	1
255.255.255.255	255.255.255.255	141.29.155.108	141.29.155.108	1

Default Gateway: 141.29.155.254

```
=====
```

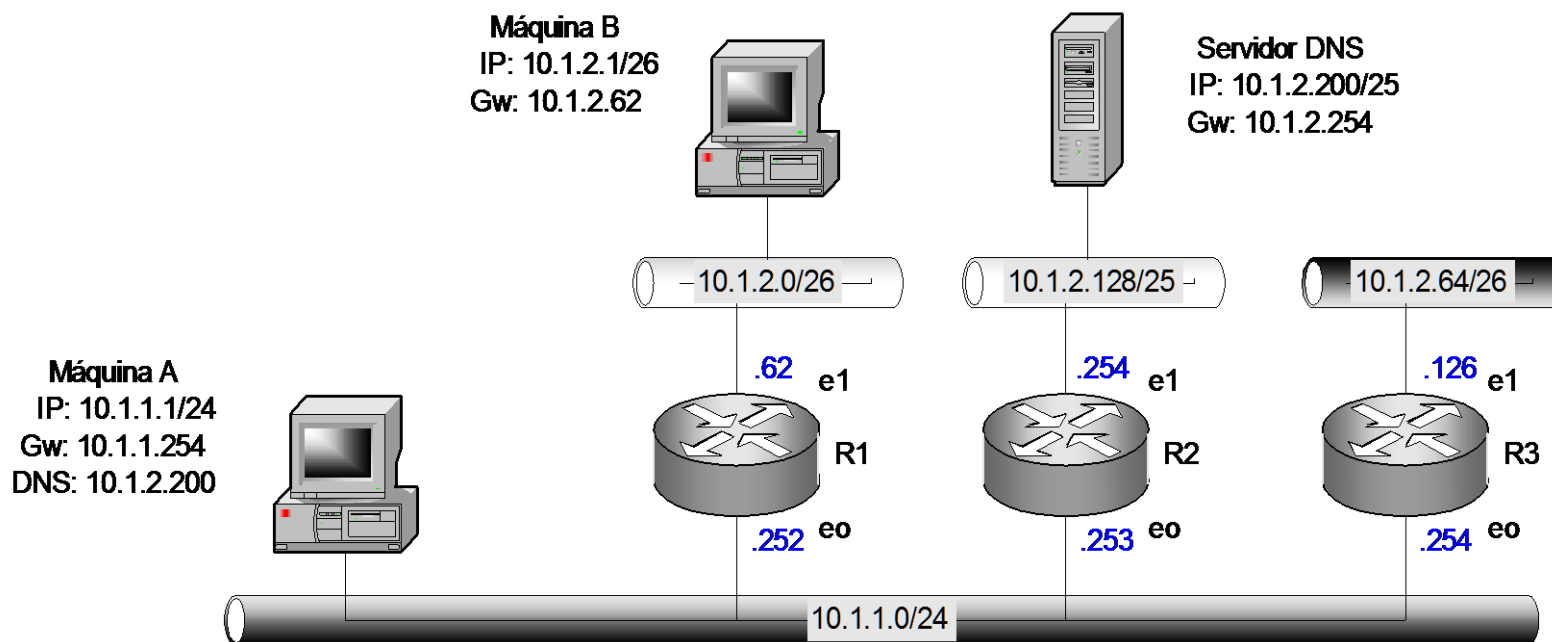
## Persistent Routes:

Network Address	Netmask	Gateway Address	Metric
200.0.0.0	255.0.0.0	141.29.155.245	3

# Exercício



- Descreva as tabelas de *routing* dos *Routers* (R1, R2 e R3)
  - Assuma que cada *router* conhece todas as redes da figura

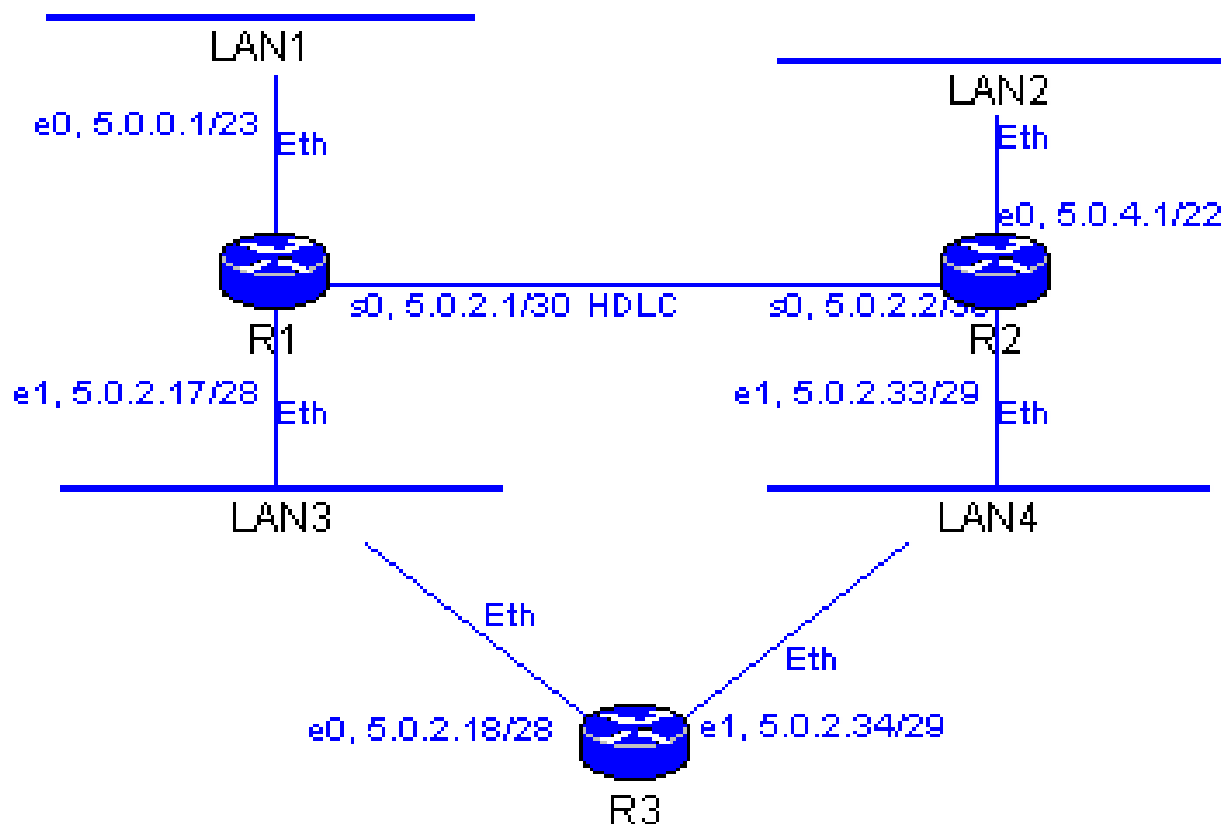




# Exercício



- Descreva as tabelas de *routing* dos *Routers* (R1, R2 e R3)
  - Assuma que cada *router* conhece todas as rede da figura



# Exercício



- A partir da tabela de encaminhamento desenhe uma topologia de rede possível

Rede	Máscara	ProximoRouter	Interface
0.0.0.0	0.0.0.0	192.168.9.254	192.168.9.227
128.1.0.0	255.255.0.0	10.1.1.1	10.1.1.9
192.1.1.226	255.255.255.254	10.1.1.2	10.1.1.9
192.168.9.224	255.255.255.224	192.168.9.227	192.168.9.227
195.16.1.224	255.255.255.240	192.168.9.252	192.168.9.227
223.0.0.0	255.0.0.0	12.18.1.1	12.18.9.27
12.18.0.0	255.255.0.0	12.18.9.27	12.18.9.27
193.137.220.0	255.255.254.0	192.168.9.253	192.168.9.227
10.1.1.0	255.255.224.0	10.1.1.9	10.1.1.9

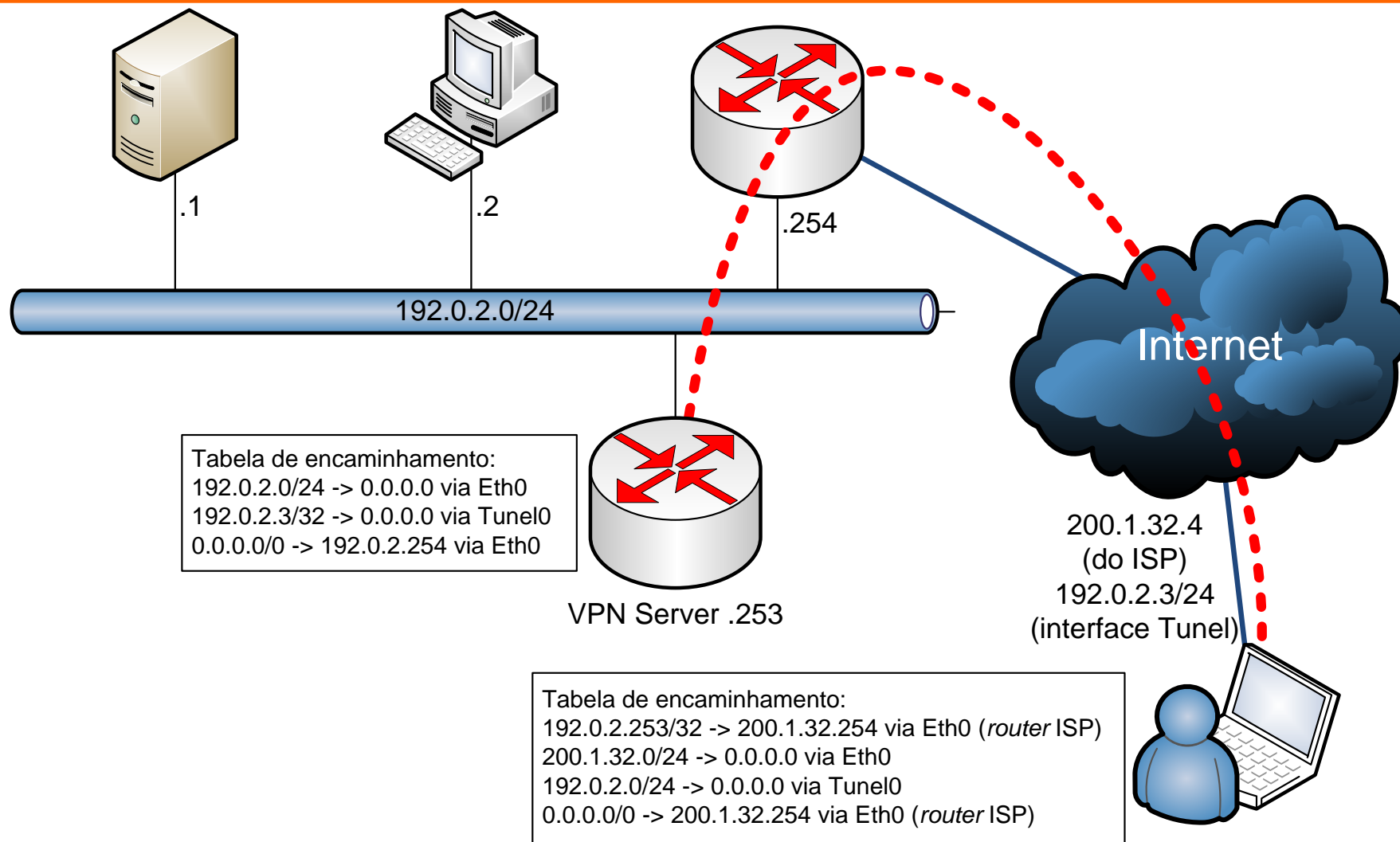
# Proxy ARP (1)

---



- Objectivo
  - Permitir que um endereço IP pertencente a uma rede se encontre fisicamente noutra
- Características
  - O *router* responde aos pedidos ARP como se da máquina em questão se tratasse
    - Normalmente tem de se activar a funcionalidade na interface do *router*
  - Apenas pode ser usado em redes que utilizam o protocolo ARP para resolução de endereços
  - Apenas funciona em máquinas com implementações de ARP que permitam ter vários endereços IP mapeados no mesmo endereço MAC
  - Documentado no RFC 1027 [1987]

# Proxy ARP (2)



# Proxy ARP (3)



- Funcionamento
  - Um utilizador liga-se via VPN ao local de trabalho e obtém um endereço IP para o “túnel” que faz parte do bloco atribuído ao segmento de rede do seu departamento
  - As máquinas do departamento não sabem à partida que aquele endereço em particular tem de sofrer uma entrega indirecta
  - Uma máquina do segmento que tentar comunicar com a do utilizador tentará fazer uma entrega directa
  - O *router* que termina a VPN do lado do departamento irá responder aos pedidos de ARP dirigidos à máquina como se de ele próprio se tratasse (actua em nome dela – *proxy*)
  - Quando recebe o datagrama a sua tabela de encaminhamento terá uma rota específica para aquele IP a apontar para o “túnel” da VPN e encaminhará até à máquina remota